

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Please amend the claims as follows:

1-8. (Cancelled)

9. (Currently Amended) A method comprising:

scanning a nonvolatile memory medium with a processor to find a first memory block containing a header indicating that the first memory block is the first memory block of an existing file stored on the nonvolatile memory medium and

finding a next memory block using a next block pointer stored in the header of the first memory block, if the existing file comprises more memory blocks than the first memory block, the existing file being opened upon completion of the finding, the header including a flag that comprises a first bit that is set if the first memory block is a first block of a file, a second bit that is set if the first memory block is a last block of the file, and a third bit that is set if the first memory block is part of a secure file.

10. (Previously Presented) The method of claim 9, further comprising repeating the finding step until either all memory blocks comprising the file have been found or an error condition occurs.

11. (Currently Amended) A method comprising:
- scanning a nonvolatile memory medium with a processor in sizes of one predetermined logical block, the nonvolatile memory medium storing an existing file;
 - for each logical block, reading a block header containing a magic number;
 - testing the magic number to determine whether the logical block is a valid block or a free block, and if the logical block is a valid block, performing a comparison of a file name encoded within the block header with a specified file name to be opened;
 - testing a flag within the block header to determine whether the logical block is the first block of the existing file, if the comparison produces a match; and
 - returning to the scanning step with the next logical block until either the comparison produces a match or all the blocks have been tested, thereby indicating an error condition, the existing file being opened if the comparison produces a match, the header including a flag that comprises a first bit that is set if the logical block is a first block of a file, a second bit that is set if the logical block is a last block of the file, and a third bit that is set if the logical block is part of a secure file.

12. (Currently Amended) A method comprising:

scanning a nonvolatile memory medium with a processor in sizes of one predetermined-sized logical block, the nonvolatile memory medium for storing a new file;

for each logical block, reading a block header containing a magic number;

testing the magic number to determine whether the logical block is a valid block or a free block, and if the logical block is a free block, modifying its block header to comprise a valid magic number, the name of the new file to be opened, and flags indicating whether the logical block is either the first block or the last block of the new file; and

returning to the scanning step with the next logical block until either the testing step has identified a free block or all the blocks have been tested, thereby indicating an error condition, the new file being opened if the testing step has identified a free block, the header including a flag that comprises a first bit that is set if the logical block is a first block of a file, a second bit that is set if the logical block is a last block of the file, and a third bit that is set if the logical block is part of a secure file.

13. (Previously Presented) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform a method, the method comprising:

scanning a nonvolatile memory medium to find a first memory block containing a header indicating that the first memory block is the first memory block of an existing file stored on the nonvolatile memory medium; and

finding a next memory block using a next block pointer stored in the header of the first memory block, if the existing file comprises more memory blocks than the first memory block, the existing file being opened upon completion of the finding, the header including a flag that comprises a first bit that is set if the first memory block is a first block of a file, a second bit that is set if the first memory block is a last block of the file, and a third bit that is set if the first memory block is part of a secure file.

14. (Previously Presented) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform a method, the method comprising:

scanning a nonvolatile memory medium in sizes of one predetermined logical block the nonvolatile memory medium storing an existing file;

for each logical block,

reading a block header containing a magic number;

testing the magic number to determine whether the logical block is a valid block or a free block, and if the logical block is a valid block, performing a comparison of a file name encoded within the block header with the specified file name to be opened;

testing a flag within the block header to determine whether the logical block is the first block of the specified file, if the comparison produces a match; and

returning to the scanning step with the next logical block until either the comparison produces a match or all the blocks have been tested, thereby indicating an error condition, the existing file being opened if said comparison produces a match, the header including a flag that comprises a first bit that is set if the logical block is a first block of a file, a second bit that is set if the logical block is a last block of the file, and a third bit that is set if the logical block is part of a secure file.

15. (Previously Presented) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform a method, the method comprising:

scanning a nonvolatile memory medium in sizes of one predetermined-sized logical block, the nonvolatile memory medium for storing a new file;

for logical block,

reading a block header containing a magic number;

testing the magic number to determine whether the logical block is a valid block or a free block, and if the logical block is a free block, modifying its block header to comprise a valid magic number, the name of the file to be opened, and flags indicating whether the logical block is either the first block or the last block of said file; and

returning to the scanning step with the next logical block until either the testing step has identified a free block or all the blocks have been tested, thereby indicating an error condition, the new file being opened if said testing step has identified a free block, the header including a flag that comprises a first bit that is set if the logical block is a first block of a file, a second bit that is set if the logical block is a last block of the file, and a third bit that is set if the logical block is part of a secure file.

16-19. (Cancelled)

20. (Previously Presented) An apparatus comprising:

means for scanning the nonvolatile memory medium to find a first memory block containing a header indicating that the first memory block is the first memory block of an existing file stored on the nonvolatile memory medium; and

means for finding a next memory block using a next block pointer stored in the header of the first memory block, if the existing file comprises more than the first memory block, the existing file being opened upon completion of said finding, the header including a flag that comprises a first bit that is set if the first memory block is a first block of a file, a second bit that is set if the first memory block is a last block of the file, and a third bit that is set if the first memory block is part of a secure file.

21. (Previously Presented) An apparatus comprising:

scanning means for scanning a nonvolatile memory medium to find a first memory block containing a header indicating that the first memory block is the first memory block of an existing file; and

finding means for finding a next memory block using a next block pointer stored in the header of the first memory block, if the file comprises more than the first memory block, the finding means operating iteratively until either all memory blocks of which the existing file is comprised have been found or an error condition occurs, the existing file being opened if all memory blocks of which the existing file is comprised have been found, the header including a flag that comprises a first bit that is set if the first memory block is a first block of a file, a second bit that is set if the first memory block is a last block of the file, and a third bit that is set if the first memory block is part of a secure file.

22. (Previously Presented) An apparatus comprising:

scanning means for scanning a nonvolatile memory medium in sizes of one predetermined logical block, the nonvolatile memory medium for storing a new file;

reading means for reading a block header containing a magic number for each logical block;

testing means for testing the magic number to determine whether the logical block is a valid block or a free block, and if the logical block is a valid block, performing a comparison of a file name encoded within the block header with a specified file name to be opened; and

checking means for checking a flag within the block header to determine whether the logical block is the first block of the existing file, if the comparison performed by the testing means produces a match, said scanning means operating iteratively on until either the comparison produces a match or all the blocks have been tested, thereby indicating an error condition, the new file being opened if the comparison produces a match, the header including a flag that comprises a first bit that is set if the logical block is a first block of a file, a second bit that is set if the logical block is a last block of the file, and a third bit that is set if the logical block is part of a secure file.

23. (Previously Presented) An apparatus comprising:

scanning means for scanning a nonvolatile memory medium in sizes of one predetermined logical block, the nonvolatile memory medium for storing a new file;

reading means for reading a block header containing a magic number for each logical block;

testing means for testing the magic number to determine whether the logical block is a valid block or a free block, and if the logical block is a free block, modifying the block header to comprise a valid magic number, the name of the file to be opened, and flags indicating whether the logical block is either the first block or the last block of the new file, the header including a flag that comprises a first bit that is set if the logical block is a first block of a file, a second bit that is set if the logical block is a last block of the file, and a third bit that is set if the logical block is part of a secure file;

the scanning means operating iteratively on next logical blocks until either the testing means has identified a free block or all the blocks have been tested, thereby indicating an error condition, the new file being opened if said testing means has identified a free block.

24. (Previously Presented) An apparatus comprising:
- nonvolatile memory medium means for storing information in the form of a file;
 - scanning means for scanning memory blocks of the nonvolatile memory medium means to find a first memory block of a selected file, the first memory block containing a header having an indicator that indicates that it is the first memory block of the selected file; and
 - finding means for finding a next memory block using a next block pointer stored in the header for selected files having more than one memory block, the selected file being opened upon completion of the finding, the header including a flag that comprises a first bit that is set if the first memory block is a first block of a file, a second bit that is set if the first memory block is a last block of the file, and a third bit that is set if the first memory block is part of a secure file.
25. (Previously Presented) The apparatus of claim 24, wherein the finding means operates iteratively until either all memory blocks of the selected file have been found or an error condition occurs.